



## Contenido

<b>Temario:</b> .....	2
<b>Plan de Pruebas Publicaciones y Comentarios</b> .....	2
<b>1. Objetivos y Alcance</b> .....	2
<b>Objetivo</b> .....	2
<b>Alcance:</b> .....	2
<b>2. Estrategia de Pruebas</b> .....	3
<b>El enfoque general será el siguiente:</b> .....	3
<b>3. Criterios de Entrada y Salida</b> .....	3
<b>Criterios de Entrada:</b> .....	3
<b>Criterios de Salida:</b> .....	3
<b>4. Recursos y Herramientas</b> .....	3
<b>5. Cronograma</b> .....	4
<b>6. Riesgos y Contingencias</b> .....	4
<b>7. Criterios de Aceptación</b> .....	5



## **Facultad de Ingeniería – TDSC – UNSTA**

**Materia:** Control de Calidad Avanzado

**Profesor:** Ing. Tulio Ruesjas Martín

**Tema:** Publicaciones y Comentarios

**Subtema:** Plan de Pruebas

**Grupo:** Bouso Gonzalo, Quevedo Diego, Perez Turbay Luciano, Sanchez Tello Rafael

**Temario:**

### 1. Red Social (Microservicios):

- Autenticación y Autorización
- Gestión de usuarios
- Publicaciones y comentarios
- Notificaciones y mensajería
- Reportes
- Estrategia, política y plan de pruebas para cada microservicio

### **Plan de Pruebas**

#### **Publicaciones y Comentarios**

##### **1. Objetivos y Alcance**

**Objetivo:** Validar que el microservicio de publicaciones y comentarios cumpla con los requisitos funcionales y no funcionales. Se buscará garantizar que todas las funcionalidades (crear, leer, actualizar, eliminar publicaciones y comentarios) operen correctamente y que la integración con otros microservicios (usuarios, notificaciones y reportes) sea estable y confiable.

**Alcance:**

- Funcionalidades CRUD para publicaciones y comentarios.
- Interacciones de usuario, como likes y respuestas.
- Integración con los microservicios de Usuarios, Notificaciones y Reportes.
- Pruebas de rendimiento, seguridad y usabilidad.



## **2. Estrategia de Pruebas**

El enfoque general será el siguiente:

- **Pruebas Funcionales:** Validar el correcto funcionamiento de las operaciones CRUD en publicaciones y comentarios.
- **Pruebas de Integración:** Asegurar que el microservicio se comuniquen efectivamente con los microservicios de Usuarios, Notificaciones y Reportes.
- **Pruebas de Comunicación Asíncrona:** Verificar que los eventos se transmitan correctamente entre microservicios utilizando RabbitMQ.
- **Pruebas de Seguridad:** Evaluar el control de acceso y la autenticación mediante tokens JWT.
- **Pruebas de Rendimiento y Escalabilidad:** Asegurar la capacidad de respuesta del sistema en escenarios de alta carga.

## **3. Criterios de Entrada y Salida**

**Criterios de Entrada:**

- Especificaciones completas de los requisitos del microservicio de publicaciones y comentarios.
- Ambiente de pruebas configurado con MongoDB, RabbitMQ y los microservicios de Usuarios, Notificaciones y Reportes.
- Acceso a herramientas de prueba (Postman, JMeter, etc.) y a la documentación de la API.

**Criterios de Salida:**

- Todas las funcionalidades CRUD y de comunicación asíncrona probadas y validadas.
- Se registran y mitigan todos los defectos críticos y de alta prioridad.
- La cobertura de pruebas cumple con los objetivos definidos para los tipos de prueba.

## **4. Recursos y Herramientas**

- **Equipo:** Testers, desarrolladores de backend y administradores de sistemas.
- **Herramientas:**
  - **Postman:** Para pruebas de API.
  - **RabbitMQ Dashboard:** Monitorización de eventos de mensajería.
  - **MongoDB Compass:** Gestión de datos de MongoDB.
  - **K6 o JMeter:** Para pruebas de rendimiento.
  - **Entorno de Pruebas:** Configuración en contenedores Docker para replicar un ambiente de producción con Kubernetes.



## 5. Cronograma

Fase	Duración	Actividades
Configuración del Entorno	1 semana	Configuración de MongoDB, RabbitMQ y herramientas
Pruebas Funcionales	2 semana	Validación CRUD en publicaciones y comentarios
Pruebas de Integración	2 semana	Pruebas con microservicios de Usuarios y Notificaciones
Pruebas de Rendimiento	1 semana	Simulación de carga y prueba de escalabilidad
Pruebas de Seguridad	1 semana	Validación de autenticación y permisos
Cierre y Documentación	1 semana	Análisis de resultados y documentación final

## 6. Riesgos y Contingencias

- **Riesgo:** Retraso en la disponibilidad del microservicio de Usuarios o Notificaciones.  
**Mitigación:** Usar datos de prueba para simular la integración.
- **Riesgo:** Falta de experiencia en RabbitMQ para gestionar la mensajería asíncrona.  
**Mitigación:** Capacitación del equipo en el uso de RabbitMQ y su dashboard.
- **Riesgo:** Exceso de carga que afecte al entorno de pruebas.  
**Mitigación:** Implementar Kubernetes para la orquestación y escalar instancias según necesidad.



## **7. Criterios de Aceptación**

- **Funcionalidad Completa:** Todas las funcionalidades de publicaciones y comentarios deben cumplir con los requisitos.
- **Criterios de Defecto:** Los defectos críticos y de alta prioridad deben resolverse antes de la entrega.
- **Desempeño y Escalabilidad:** El microservicio debe manejar un nivel definido de carga sin comprometer la experiencia del usuario.